

PCT/DE2004/001379  
2003P08340WOUS

- 1 -

1 Description

2

3 Method for generating a structure representation which  
4 describes a specific automation system

5

6 To control or monitor automated processes, what are known as  
7 automation systems are normally used today. The automated  
8 processes may be process-engineering processes, automated  
9 production methods or else generation and distribution systems  
10 for electrical power, for example. An automation system  
11 normally comprises field transmitters which are connected to  
12 the automated process, which are arranged in proximity to the  
13 process and which use suitable sensors, such as current and  
14 voltage converters, flow meters or concentration measuring  
15 instruments, to obtain particular measured data from the  
16 process (e.g. currents, voltages, mass flows, concentrations).  
17 On the basis of these measured data, the process can be  
18 monitored and controlled. By way of example, the measured data  
19 can be forwarded to suitable output devices, such as screen  
20 displays, and can be shown there, for example in the form of  
21 graphics or tables, to the operating personnel for the  
22 respective process. If the field transmitters are what are  
23 known as electrical protective devices then the captured  
24 measured data are normally checked automatically for observance  
25 of particular operating parameters and, if the prescribed  
26 operating parameters are not observed, a suitable protective  
27 measure is automatically taken, such as opening a circuit  
28 breaker in an electrical power distribution system.

29

30 Normally such automation systems are controlled by data  
31 processing devices which can use

1 a suitable piece of control software to perform all the steps  
2 required for control and monitoring. To this end, such a data  
3 processing device and the relevant control software need to be  
4 matched, inter alia, precisely to the structure or design of  
5 the respective specific automation system for the relevant  
6 process which has to be automated. Such matching is today  
7 usually carried out in the actual programming stage of the  
8 relevant control software for the data processing device in a  
9 respective specific automation system, which means that the  
10 programmers and developers of this control software actually  
11 need to know the design of the specific automation system at  
12 the time of programming.

13  
14 By way of example, German laid-open specification DE 100 53 665  
15 A1 discloses a process control system or an automation system  
16 for remotely monitoring and controlling process-engineering  
17 processes. In the case of the process control system known from  
18 the laid-open specification, the measured data are displayed  
19 and the process control system is operated using a process  
20 display which is stored to a central location in the data  
21 processing device and which contains the specific structure of  
22 the automation system for the relevant process. This process  
23 display has been created manually and stored at the central  
24 location in the data processing device before the automation  
25 system is actually started up. In the event of any changes to  
26 the automation system, this process display also needs to be  
27 matched accordingly.

28  
29 The invention is based on the object of specifying a method  
30 which provides a comparatively simple way of

1 producing a piece of control software for a data processing  
2 device in a specific automation system.

3  
4 The invention achieves this object by means of a method for  
5 generating a structure representation which describes a  
6 specific automation system from a model structure  
7 representation which describes a general automation system,  
8 where the model structure representation has a structured  
9 representation of functional groups in the general automation  
10 system and their links to one another, and each functional  
11 group can be assigned one or more components of the specific  
12 automation system, and where the following steps are performed:

- 13 - a text file reproducing the model structure representation  
14 is provided for a data processing device which controls  
15 the specific automation system;  
16 - those components of the specific automation system which  
17 can be jointly assigned to a functional group in the model  
18 structure representation are ascertained by the data  
19 processing device, and  
20 - the ascertained components are entered into the model  
21 structure representation to generate the structure  
22 representation which describes the specific automation  
23 system.

24  
25 The fundamental advantage of the inventive method is that the  
26 software for the data processing device which controls the  
27 specific automation system does not need to be manually matched  
28 individually to the specific automation system. The reason for  
29 this is that the inventive method involves only a model  
30 structure representation which describes a general automation  
31 system being provided in the software at the time of  
32 programming, said model structure representation being  
33 automatically matched to the specific

1 automation system only when the automation system is started  
2 up.

3

4 In other words, a general piece of control software is provided  
5 containing a model which applies to the design of various  
6 automation systems and which, when a specific system is started  
7 up, is independently - that is to say without any manual  
8 settings - matched to the design or structure thereof.

9

10 Another advantage of the inventive method is that this allows  
11 the creation of a general piece of control software which can  
12 be matched to a multiplicity of different automation systems.  
13 There is thus no separate creation of a respective piece of  
14 control software for every single instance of various  
15 automation systems.

16

17 In one advantageous development of the inventive method,  
18 instructions contained in the text file prompt the data  
19 processing device to check only selected functional groups to  
20 determine whether a plurality of components of the specific  
21 automation system can be jointly assigned to this functional  
22 group. In this way, the model structure representation can be  
23 matched to the structure representation of the specific  
24 automation system particularly quickly and efficiently, since  
25 it is actually possible to stipulate in advance those  
26 functional groups for which a plurality of components of the  
27 automation system can be assigned in the first place. There is  
28 thus no checking of the other functional groups by the data  
29 processing device.

1 In another advantageous embodiment of the inventive method, the  
2 data processing device ascertains those components of the  
3 specific automation system which can be jointly assigned to a  
4 functional group in the model structure representation by  
5 sending an electronic query to the respective components or to  
6 a common control device which is superordinate to them, and the  
7 respective components or the common control device which is  
8 superordinate to them respond(s) to this electronic query by  
9 sending an electronic response to the data processing device  
10 with an identification key which is respectively unique for  
11 them. In this way, the plurality of components which can be  
12 assigned to a common functional group can be identified  
13 particularly easily, since the data processing device can do  
14 this merely by asking the multiple instances of components for  
15 their unique identification keys. The respective unique  
16 identification key can be used to distinguish each component of  
17 the specific automation system clearly from all other  
18 components. Identification keys within this context may be slot  
19 numbers, unique addresses, component identification numbers,  
20 component names arising once within the specific automation  
21 system and product or serial numbers for individual components,  
22 for example.

23  
24 In this context, the electronic query within the scope of the  
25 invention may also be sent to a common control device - usually  
26 a software module - which is superordinate to the respective  
27 components of the specific automation system and which controls  
28 the response from the respective identification key if more  
29 than one component can be assigned to the relevant functional  
30 group.

1 In this connection, it is also considered to be advantageous if  
2 components of the specific automation system which can each be  
3 jointly assigned to a functional group in the model structure  
4 representation use identification keys of the same type when  
5 sending the electronic response. In this way, the relevant  
6 components can be distinguished particularly easily, since  
7 although all the components of the specific automation system  
8 which can be assigned to a functional group have different  
9 values of identification keys, they generally use the same type  
10 of identification key. By way of example, in this context the  
11 identification key used is a slot number which can be used for  
12 identifying the respective components clearly from their  
13 different identification key values, that is to say in this  
14 case different slot numbers.

15  
16 In addition, it is considered to be an advantageous development  
17 of the inventive method if the respective components  
18 additionally send further data characterizing them with the  
19 electronic response. In the case of this development, specific  
20 further information about the relevant components of the  
21 automation system can advantageously be picked up by the data  
22 processing device when the specific automation system is  
23 actually started up. Examples of such further information may  
24 be a more precise description of the relevant component, its  
25 respective status (on, off, fault) or, by way of example, the  
26 formats of the data which this component needs to send to the  
27 data processing device and receive while the automation system  
28 is operating.

1 It is considered to be a further advantageous refinement of the  
2 inventive method that even a functional group which can be  
3 assigned a single component has this component ascertained and  
4 entered into the model structure representation to complete the  
5 structure representation which describes the specific  
6 automation system. In this way, the full automation system with  
7 all the components can be identified and entered into the  
8 structure representation actually within the startup described  
9 above. In line with this development, components which are the  
10 only ones to be able to be assigned to a particular functional  
11 group in the model structure representation are thus also  
12 additionally identified and entered into the model structure  
13 representation.

14  
15 Particularly advantageously, the respective components of the  
16 specific automation system can be addressed and identified by  
17 the data processing device if the respective components of the  
18 specific automation system are addressed by the data processing  
19 device using a component path which contains at least one  
20 identification for the respective component. In this context, a  
21 component path can be constructed in similar fashion to file,  
22 directory or device paths which are known from operating  
23 systems for home and office computers.

24  
25 It is also considered to be advantageous if the data processing  
26 device ascertains information which is typical of a component  
27 of a functional group or which is common to a plurality of  
28 components of a functional group by generating a type path  
29 which indicates the relevant functional group from the relevant  
30 component path, and the data processing device uses this type  
31 path  
32

1 to read the information for the relevant functional group from  
2 the text file. In this way, it is merely possible to use the  
3 component path corresponding to a respective component to  
4 ascertain information which is typical of this component from  
5 the relevant functional group following conversion into a type  
6 path. Such information may be, by way of example, the type of  
7 components which can be assigned to this functional group, and  
8 formats for the data which are to be interchanged with these  
9 components.

10  
11 A type path indicating the relevant functional group can be  
12 generated by the data processing device particularly easily if  
13 the data processing device generates the type path from the  
14 component path by removing the at least one identification for  
15 the respective component from the component path to form the  
16 type path.

17  
18 In this connection, it is also regarded as being particularly  
19 advantageous that the component path and the type path are  
20 formulated using the language XPath. The language XPath is a  
21 language for navigation and addressing particularly in XML  
22 documents which has been normalized by the international  
23 standardization committee W3C. More detailed information on  
24 XPath can be found at <http://www.w3.org/TR/xpath>.

25  
26 Furthermore, in another advantageous embodiment of the  
27 inventive method, the structure representation which describes  
28 the specific automation system is converted into a graphical  
29 representation by the data processing device. Such a graphical  
30 representation can be used by an operator of the automation  
31 system to  
32

1 obtain a fast and comprehensive overview of the full automation  
2 system.

3  
4 It is regarded as particularly advantageous in this connection  
5 if the graphical representation based on the structure  
6 representation is displayed using a user device which belongs  
7 to the data processing device. In this context, an operator of  
8 the automation system can particularly advantageously obtain an  
9 overview of the precise design of the specific automation  
10 system, for example using a user device connected to a central  
11 computer in the data processing device, such as a control  
12 station or a laptop.

13  
14 In this context, it may also be advantageous if the structure  
15 representation is converted into the graphical representation,  
16 and displayed, using a browser device on the user device. In  
17 this way, a user device can obtain access to the graphical  
18 representation of the structure of the specific automation  
19 system particularly easily in an Internet-based automation  
20 system, for example, because a browser device can be used for  
21 access which is largely independent of hardware and operating  
22 system.

23  
24 It is also regarded as particularly advantageous that XML is  
25 used for the text file reproducing the model structure  
26 representation. The XML (Extended Markup Language) format is  
27 particularly well suited to describing hierarchically  
28 structured systems. For this reason, it can also be used with  
29 particular advantage to represent a normally hierarchically  
30 structured design for an automation system. The XML format is  
31 also system-independent, that is to say can

1 be edited data processing devices with various operating  
2 systems using different programming languages.

3  
4 To explain the inventive method in more detail,

5  
6 Figure 1 shows a schematic illustration of an automation  
7 system in a block diagram,

8  
9 Figure 2 shows an exemplary embodiment of a structure  
10 representation which describes a specific automation  
11 system,

12  
13 Figure 3 shows an exemplary embodiment of a model structure  
14 representation which describes a general automation  
15 system, and

16  
17 Figure 4 shows an exemplary embodiment of a user display for  
18 operating an automation system.

19  
20 Figure 1 uses a type of block diagram to show one possible  
21 design for an automation system 1 by way of example. An  
22 automated process (not shown in figure 1) involves field  
23 transmitters 2A to 2D, which are connected to the automated  
24 process via sensors and/or converters (likewise not shown in  
25 figure 1). The process may be a process-engineering process or  
26 an automated production process, for example. In the text  
27 below, however, it will be assumed that the process is a power  
28 supply system, for

1 example a power supply mains. In this case, the field  
2 transmitters connected to the power supply system are control  
3 appliances or electrical protective appliances for monitoring  
4 and for protecting the power supply mains, for example. The  
5 field transmitters 2A to 2D deliver measured data from the  
6 process, for example voltage and current measurements, to a  
7 data processing device 4 via a bus system 3. In line with  
8 figure 1, the data processing device 4, which is set up to  
9 control the automation system 1, comprises a central computer 5  
10 and user devices 6A to 6D, which communicate with the central  
11 computer 5 by means of various options. For example, the user  
12 device 6A, which may be a local control station, for example,  
13 is hardwired directly to the central computer 5. The user  
14 devices 6B and 6C, e.g. office or control room computers, are  
15 connected to the central computer 5 via a network 7, for  
16 example the Internet or an Intranet, and the user device 6D,  
17 e.g. a laptop, is wirelessly connected to the central computer  
18 5 by means of combined transmission and reception devices 8A  
19 and 8B.

20  
21 Figure 2 uses a tree structure to show an exemplary embodiment  
22 of a design for a (specific) automation system of this type  
23 with a plurality of components. In this context, a central  
24 processor unit (CPU) 20, which may be held in the central  
25 computer 5 shown in figure 1, for example, first of all  
26 controls an integral driver device 21 which is used to regulate  
27 the communication between the CPU 20 and the remaining  
28 components of the automation system, for example. The integral  
29 driver 21 can also retrieve an information block 22. By way of  
30 example, information contained in the information block 22 can  
31 contain a version number and a production

32

1 date for the integral driver 21. In line with figure 2, the  
2 integral driver 21 in turn controls four "single drivers" 23A  
3 to 23D, which in turn have particular appliances 24A to 24E  
4 subordinate to them. It can be seen that the fourth single  
5 driver 23B at this location has two associated appliances 24D  
6 and 24E. In a subsequent structure plane, the appliances 24A to  
7 24E finally have subordinate sensors 25A to 25F which are  
8 connected to the process.

9  
10 It is also possible to see in figure 2 that most components of  
11 the specific automation system shown in figure 1 correspond to  
12 particular blocks in the structure representation shown in  
13 figure 2. In this context, some blocks in figure 2 represent  
14 clear components of the specific automation system, for example  
15 appliances 24A to 24E and sensors 25A to 25F. Others represent  
16 software components, for example single drivers 23A to 23D or  
17 the integral driver 21. The information block 22 has no actual  
18 depiction in the specific automation system and in this context  
19 serves merely to give a structure to data and information  
20 within the structure representation and hence serves to provide  
21 them with better clarity. Said software components and the  
22 information block 22 will normally be produced on the data  
23 processing device 4 (cf. figure 1), which is indicated in  
24 figure 2 by a dashed frame.

25  
26 For the automation system to operate, it is necessary for there  
27 to be such a structure representation of the specific  
28 automation system with its relevant components for a piece of  
29 control software for the data processing device 4 shown in  
30 figure 1. In line with the invention, such a

31

1 structure representation is obtained from a model structure  
2 representation, as shown in figure 3, for example.

3  
4 Figure 3 shows such a model structure representation by way of  
5 example, said model structure representation being able to be  
6 applied generally to a large number of automation systems. In  
7 this context, instead of individual components of a specific  
8 automation system the figure shows only their functional  
9 groups, that is to say classes of components in a fictitious  
10 automation system, so to speak. This is identified in the  
11 illustration shown in figure 3 by the abbreviation "FG" in the  
12 top left-hand corner of each individual block. In line with  
13 figure 3, the topmost structure plane holds a functional group  
14 (FG) CPU 31 which is superordinate to an FG "integral driver"  
15 32. The FG "integral driver" 32 is in turn superordinate to  
16 functional groups "single driver" 33 and "information" 34 which  
17 are arranged parallel to one another. Finally, the FG "single  
18 driver" 33 has a subordinate FG "appliance" 35 and the latter  
19 has a subordinate FG "sensor" 36. A model structure  
20 representation of this type can be used generally to describe  
21 the design of a multiplicity of automation systems.

22  
23 To generate a structure representation of a specific automation  
24 system from the model structure representation shown in figure  
25 3, for example in a similar manner to figure 2, it is necessary  
26 to ascertain the true number of components which can be  
27 assigned to the respective functional groups, particularly at  
28 the locations indicated by asterisks 37 in figure 3. By way of  
29 example, the functional group "single driver" can be assigned a  
30 plurality of single driver components in a specific automation  
31 system. In addition, the individual components of the specific  
32 automation system need to be entered into the

33

1 model structure representation, in order to obtain a  
2 corresponding structure representation therefrom.

3  
4 The procedure which is to be applied for this purpose will be  
5 described in more detail below:

6  
7 To match a piece of control software to a specific automation  
8 system, a model structure representation, for example as shown  
9 in figure 3, is first of all required in the form of a text  
10 file. Such a text file can be created particularly conveniently  
11 in XML (Extended Markup Language) format, since XML can be used  
12 particularly well for describing hierarchically structured  
13 systems. An example of such a text file in a shortened XML  
14 version is indicated below:

15  
1 <CPU>  
2 <Integral driver>  
3 <Information>  
4 <Driver number type = "int"/>  
5 <Date of creation type = "string"/>  
6 </Information>  
  
7 <Single driver ResolveCardinalities = "true" Key = "Driver  
No.">  
8 <Status type = "string"/>  
9 <Designation type = "string"/>  
10 <Appliance ResolveCardinalities = "true" Key = "Appliance  
No.">  
11 <Status type = "string"/>  
12 <Designation type = "string"/>  
:  
:  
:  
:  
21 <Sensor Resolve Cardinalities = "true" Key = "Sensor No.">  
:

PCT/DE2004/001379  
2003P08340WOUS

- 14a -

:

:

31 </Sensor>

32 </Appliance>

33 </Single driver>

34 </Integral driver>

35 </CPU>

1

2

1 Such a text file can be used to describe the model structure  
2 representation shown in figure 3. By way of example, the text  
3 file shows the individual functional groups "CPU", "Integral  
4 driver", "Information" etc., which have further associated data  
5 when required. By way of example, the functional group  
6 "Information" may contain data indicating a driver number or a  
7 date of creation. This is indicated in lines 4 and 5 of the  
8 text file shown. In addition, the respective data type used is  
9 also indicated, for example the data type "integer" (int) is  
10 used for the driver number, and the date of creation is in the  
11 "string" type. In line 7 the text file also shows the  
12 instruction "ResolveCardinalities = true", which means that at  
13 this location in the functional group "Single driver" it would  
14 be possible to assign not just a single component but rather a  
15 plurality of single driver components of the specific  
16 automation system jointly. To distinguish between the  
17 individual single driver components of the specific automation  
18 system, the expression Key = "Driver No." is used to stipulate  
19 the addressing number, for example, of a single driver  
20 component as a unique identification key. The same applies to  
21 the functional groups "Appliance" and "Sensor" with the  
22 identification keys appliance number (appliance No.) and sensor  
23 number (sensor No.).

24  
25 A text file in such or similar a form needs to be made  
26 available to the data processing device which controls the  
27 specific automation system. By way of example, it is  
28 transmitted to the data processing device after it has been  
29 created on a programming workstation. However, one particular  
30 advantage in the use of such a model structure representation  
31 is that a single model structure representation in the form of  
32 a text file can be copied a plurality of times and can be used  
33 for a plurality of

34

1 automation systems. The development and programming involvement  
2 in advance is thus significantly reduced.

3  
4 The data processing device in the specific automation system  
5 then needs to be used to convert the model structure  
6 representation into a structure representation which is matched  
7 to the specific automation system. To this end, the data  
8 processing device checks the text file for instructions such as  
9 "ResolveCardinalities = true", for example, in order to  
10 ascertain those locations at which there may be a plurality of  
11 components which can be assigned to a functional group. At  
12 these locations, the data processing device requests the  
13 relevant components in the specific automation system, for  
14 example by virtue of the data processing device transmitting an  
15 electronic query containing the respective type of an  
16 identification key which has been sought for the specific  
17 functional group. This electronic query is respectively  
18 answered by the components which can be assigned to this  
19 functional group by virtue of them sending their identification  
20 key, e.g. their appliance number, to the data processing device  
21 as an electronic response.

22  
23 Alternatively, it is also possible for all or some similar  
24 components of the specific automation system to have a super-  
25 ordinate common control device which receives the electronic  
26 query and coordinates the responses of the respective  
27 components to the data processing device.

28  
29 Together with the identification key, it is also possible to  
30 send further data characterizing the respective component, for  
31 example, such as a more accurate designation for the component  
32 or the status of the component (e.g. on, off,

33

1 fault). The data processing device enters the returned  
2 components into the text file using their identification keys  
3 and checks the further structure planes of the text file in  
4 similar fashion.

5  
6 At all locations at which there are instructions to check the  
7 specific automation system for a plurality of components which  
8 can be jointly assigned to a functional group, the data  
9 processing device carries out the method which has just been  
10 described. In this way, all components of the automation system  
11 which can be jointly assigned to a respective functional group  
12 are incorporated into the text file, so that ultimately a  
13 structure representation of the specific automation system,  
14 likewise in text form, is produced, as indicated again in  
15 shortened version and in XML format below by way of example:

16  
1 <CPU>  
2 <Integral driver>  
:  
:  
:  
11 <Single driver Key = "Driver No." Driver No. = "100">  
:  
:  
:  
21 <Appliance Key = "Appliance No." Appliance No. = "1">  
:  
:  
:  
31 <Sensor Key = "Sensor No." Sensor No. = "1001">  
:  
:  
:  
41 </Sensor>  
42 <Sensor Key + "Sensor No." Sensor No. = "1002">

PCT/DE2004/001379  
2003P08340WOUS

- 17a -

:  
:  
:

51 </Sensor>

52 </Appliance>

53 <Appliance Key = "Appliance No."    Appliance No. = "2">

```
:  
:  
:  
61  </Appliance>  
:   </Single driver>  
:  
:  
71  </Integral driver>  
72  </CPU>
```

1  
2 From the structure representation shown for the specific  
3 automation system in text format, it can be seen that  
4 individual components, such as a single driver with the driver  
5 number 100, are entered below the functional group "Integral  
6 driver", for example. The single driver with driver No. 100 has  
7 subordinate appliances with the appliance numbers (appliance  
8 No.) 1 and 2 in the structure representation shown. Further  
9 appliances may be listed in a similar manner. The appliance  
10 with appliance No. 1 in turn has subordinate sensors with the  
11 sensor numbers 1001 and 1002. The structure explained  
12 corresponds to the left-hand branch of the tree structure of  
13 the specific automation system with the first single driver  
14 23A, and the first appliance 24A and the sensors 25A and 25B  
15 which is shown in figure 2, for example. The other branches of  
16 the structure representation shown in figure 2 can be  
17 transferred to the structure representation in text format in  
18 similar fashion.

19  
20 In addition to the respective identification keys, such as  
21 driver numbers, product numbers and sensor numbers, of the  
22 individual identified components, it is also possible for  
23 further data describing the components to be incorporated into  
24 the structure representation. Thus, by way of example, this may  
25 contain information indicating a status for the respective  
26 component and a more accurate designation for the component.

27

1 In addition to such components as can be jointly assigned to a  
2 functional group in the model structure representation, the  
3 other components of the specific automation system can also be  
4 entered into the model structure representation in the same  
5 cycle to form a complemented structure representation of the  
6 specific automation system. In the case of these components,  
7 precisely one component is therefore assigned to precisely one  
8 functional group. In the case of figures 2 and 3, this applies  
9 to the CPU 20 and the functional group "CPU" 31, for example.  
10 These other components can be detected in similar fashion to  
11 the procedure described with an electronic query from the data  
12 processing device and a corresponding electronic response from  
13 the respective component, where, by way of example, a value for  
14 an identification key for the respective component is  
15 transmitted to the data processing device and is entered into  
16 the model structure representation. In similar fashion to the  
17 method described above, it is possible, in this case too, to  
18 send further data identifying the respective component in  
19 addition to the identification keys. In this way, a full  
20 structure representation of the specific automation system with  
21 all the available components is finally obtained from the model  
22 structure representation.

23  
24 To designate a component of the specific automation system and  
25 possibly to request information or measured data from the  
26 latter or to transmit it to the latter, the data processing  
27 device needs to use unique addressing for the respective  
28 component. In this context, it is appropriate to use what are  
29 known as component paths - or else entity paths - in which, in  
30 similar fashion to the form which is used by known office and  
31 home computer operating systems, the respective components are  
32 listed separated by

33

1 oblique strokes. It is important that to address a specific  
2 component of the specific automation system the respective  
3 identification keys which are unique for the sought component  
4 also need to be incorporated into the component or entity path  
5 in addition.

6  
7 The text below shows an example of a component path for  
8 addressing the first appliance 24A (cf. figure 2) and for  
9 requesting its status:

10  
11 "CPU/Integral driver/Single driver['Driver No. = 100']/  
12 Appliance['Appliance No. = 1']/Status"

13  
14 Organization and navigation using such paths can be carried out  
15 particularly conveniently using the XPath language normalized  
16 by the standardization committee W3C, for example. Further  
17 information relating to XPath can be retrieved on the Internet  
18 at the URL <http://www.w3.org/TR/xpath>.

19  
20 In addition, it may be advantageous not just to address a  
21 specific component using an appropriate component path, but  
22 rather to obtain general information about said component's  
23 respective functional group. This information is contained in  
24 the model structure representation. Information about a  
25 functional group may be "metadata", for example, which indicate  
26 a data type in which data can be received or sent by components  
27 of the specific automation system which are associated with the  
28 respective functional group. When a component path as indicated  
29 above is used, it is a particularly simple matter to generate a  
30 "type path" for retrieving such information about the  
31 individual functional groups. The reason for this is that this  
32 simply requires all parts of the component path

33

1 which respectively contain the specific identification keys of  
2 components of the specific automation system to be removed; in  
3 the example above, the type path remaining is as follows, for  
4 example:

5  
6 "CPU/Integral driver/Single driver/Appliance/Status"

7  
8 This type path can now be used to request general information  
9 about the file format, for example, in which the status of the  
10 individual components associated with the functional group  
11 "Appliances" is output. This indicated type path refers to line  
12 11 of the text file indicated above, for example, in which the  
13 data type indicated for the status of the functional group  
14 "Appliance" is the type "string".

15  
16 Using the structure representation produced in this manner for  
17 the specific automation system, a user of the automation system  
18 can also access specific components of the automation system  
19 very easily. To this end, a display device or an external  
20 computer with a display device can be connected to the central  
21 computer in the data processing device, for example, and can  
22 allow navigation in the automation system in a browser window,  
23 for example, as shown by way of example in figure 4.

24  
25 Figure 4 shows a browser window 41 in which, besides other  
26 information, particularly a structure representation of the  
27 specific automation system ("System 1") is shown in a graphical  
28 representation as a structure tree 42 by way of example. Using  
29 such a (graphical) structure representation of the specific  
30 automation system, a user of the automation system can, by way  
31 of example, access the sixth sensor - as indicated by the  
32 structure tree 42 -

33

1 (cf. 25F in figure 2) and can retrieve particular information,  
2 such as measurements or status values, from the sensor. This  
3 information can then be displayed in a further window 43 in the  
4 form of a table, for example. The graphical representation of  
5 such a structure representation describing the specific  
6 automation system can be generated comparatively easily from an  
7 XML file as indicated above. This can be converted either  
8 directly by the central computer of the data processing device  
9 or by the browser device of a user computer connected to the  
10 central computer of the data processing device, for example.  
11